INTERACTIVE SCULPTING FOR VOLUMETRIC EXPLORATION AND FEATURE EXTRACTION

This application is a continuation of PCT/CA00/00043 filed on January 21, 2000.

The present invention relates to the field of three-dimensional (3-D) image rendering. More specifically, the present invention provides a system and method of interactive sculpting for volumetric exploration and feature extraction.

Background of the Invention

Volume visualization has become increasingly important in clinical practice for the display and analysis of volumetric datasets acquired with imaging methods such as Computed Tomography (CT) or Magnetic Resonance Imaging (MRI). Benefits of volume visualization include the ability to obtain oblique views, the increased understanding of complex geometric structures; and the ability to measure volumes, areas, and distances. Volume visualization also provides the ability to explore the spatial relationship between an organ and its surrounding structures or tissues. There are two main classes of volume visualization techniques; surface rendering and volume rendering. The present invention relates to volume rendering which is a technique that generates a two-dimensional (2-D) projection directly from the 3-D volume data without requiring any intermediate geometrical data structure.

Many different approaches to volume rendering have been proposed over the years. However, they all suffer to some extent from the problem that volume rendering algorithms are computationally and memory intensive, although, ever-increasing processor speed, affordable memory, and clever optimizations have made it possible to perform volume rendering on low-cost, general purpose personal computers. However, even with interactive rendering speed, the exploration and visualization of a volumetric dataset is a challenging task. The increasing amount of data produced by the imaging devices and the occlusion of the structures of interest by other portions of the subject call for flexible methods to focus the visualization to a very specific part of the dataset.

The classical approach to address these issues has been 3-D anatomical feature extraction by segmentation. These methods tend to be very time consuming (i.e. slice by slice tracing) or very anatomy and/or imaging modality specific. A number of researchers have proposed a

more general semi-automatic segmentation method, however the binary decision process imposed by the segmentation method often results in loss of information and reduced image quality. See, for example, Schiemann et al., "Interactive 3D Segmentation", Visualization in Biomedical Computing 1992, SPIE Vol. 1808, pp. 376-383, which is incorporated herein by reference.

An alternative approach is to remove regions with no diagnostic value or regions that obscure the structure of interest, rather than extracting the structure itself. The classical axis-aligned cut plane model is often used for this purpose but in many cases, when complicated morphologies are present, the results are far from optimal. Pflessor et al., "Towards Realistic Visualization for Surgery Rehearsal" First International Conference of Compute Vision, Virtual Reality and Robotics in Medicine (CVR Med '95), pp. 487-491, April 1995, which is incorporated herein by reference, presents a volume sculpting technique with application to surgery rehearsal. However, in this technique a segmentation step is required prior to data manipulation. The technique introduced sculpting tools that perform free-form cutting of pre-segmented objects. Wang and Kaufman "Volume Sculpting", 1995 Symposium on Interactive 3D Graphics, pp. 157-156, April 1995, which is incorporated herein by reference, provides a similar technique but with the focus on a modeling tool for the creation of a three-dimensional object, not on the extraction of specific features from a volume dataset.

It is an object of the present invention to provide a system and method of volume rendering which overcomes the deficiencies of the prior art.

Summary of the Invention

Accordingly, in a first aspect, the present invention provides a system of extracting a visual feature from a dataset, comprising a storage means for storing said dataset; retrieval means for retrieving said dataset from said storage means; display means for displaying an image of said retrieval dataset; means for defining a block of voxels corresponding to a selected portion of said displayed dataset, said block containing said visual feature therein; means for removing from said block voxels not containing said visual feature, to generate a feature block; and means for generating a mask from said feature block; means for rendering said dataset using said mask..

Interactive sculpting is an advanced technique for volumetric exploration and features extraction in scalar datasets. Used in conjunction with known fast volume rendering techniques, the present technique proves to be a powerful and flexible tool especially for a modality like magnetic resonance imaging (MR), where the underlying physics is not compatible with the use of opacity transfer functions in volume rendering for tissues separation. The technique of the present invention is substantially different from classical segmentation approaches that require a precise definition of the volume of interest. In the present approach, the sculpted volume does not have to be defined precisely; all the details are extracted by the volume renderer during the projection.

In one aspect, the present invention defines two sculpting tools; a jigsaw and an interpolator.

The jigsaw tool allows the user to "saw-off" parts of the volume, i.e., remove regions that are not of interest. The tool is flexible in that it can be applied at any viewing angle and any shape can be drawn which may either include or exclude the region of interest.

The interpolator tool allows the user to roughly outline the region of interest on a few image slices and the tool then generates the volume of interest by interpolating between these slices. The inputs to this tool are inexact regions of interest obtained from manual contouring on 2-D images.

For both tools, the location and orientation information required by the reconstruction process can be obtained directly from the display.

In accordance with this invention there is provided a method for extracting a visual feature from a dataset representing an image of an object, said method comprising the steps of: displaying said dataset; defining a block of voxels corresponding to a selected portion of said displayed dataset, said block containing said visual feature therein; removing from said dataset voxels not contained by said selected portion, to generate a feature block; generating a mask from said feature block; and rendering said dataset using said mask.

Brief Description of the Figures

Embodiments of the present invention will now be described, by way of example only, with reference to the following figures, in which:

Figure 1 is a schematic overview of a system in accordance with the present invention;

Figure 2 is an illustration of the use of a jigsaw tool in accordance with the present invention;

Figure 3 is an illustration of the use of an interpolator tool in accordance with the present invention;

Figures 4(a) and **4(b)** are examples of images generated by the system of the present invention;

Figures 5(a) and **5(b)** are examples of images generated by the system of the present invention;

Figure 6 is a schematic representation of a volume rendering pipeline;

Figure 7 is a representation of the pipeline used to create the images of Figure 4(a) and 4(b); and

Figure 8(a) is a flow chart showing the use of jigsaw tool;

Figure 8(b) is a flow chart showing the use of the interpolator tool;

Figure 8(c) is a flow chart showing a craniotomy on a MR study according to an embodiment of the invention; and

Figure 8(d) is a flow chart showing an identification of a pathology in a CTA study.

Detailed Description of a Preferred Embodiment

A system in accordance with the present invention is shown generally at 10 in Figure 1. System 10 comprises a processor 12 connected to a user input device 14. User input device 14 may be, for example, a keyboard, a mouse or a touch-screen display. Processor 12 is also connected to a data storage unit 18, and to display means 22.

Data storage unit 18 is used to store pre-acquired 2-D image slices of a subject in digital form. The stacks of slices are generally referred to as a dataset. The 2-D image slices are represented by dashed lines in Figure 1. The 2-D image slices may be acquired using any

imaging technique, including but not limited to Computed Tomography, Magnetic Resonance Imaging, and Ultrasound.

Display means 22 displays a rendered volume 26 which is reconstructed from the pre-acquired 2-D image slices stored in data storage unit 18. The rendered volume displayed in Figure 1 has a "cut-surface" 28 and a defined "volume of interest" (VOI) 30 which will be explained below.

Prior to describing the method of volumetric exploration and feature extraction of the present invention, the two primary tools of the present invention will be described.

(i) The Jigsaw

The jigsaw is a versatile cutting tool. The jigsaw can be used in a number of different ways. The simplest way is to reduce the effective size of the bounding box by sawing off rectangular regions. In practice, the jigsaw can be used to quickly remove structures that obscure the anatomy of interest. This method is efficient because careful outlining of unwanted structures is not required. It also plays an important role in fast exploration of multiple datasets.

When reducing the size of the bounding box is not adequate for removing unwanted anatomy one could use the jigsaw to generate a cylindrically extruded mask, whose cross section is defined by an arbitrary 2-D shape. This tool is not restricted to three orthogonal views. In fact, the direction of extrusion is normal to the plane on which the 2-D shape is defined. In other words, this tool can be applied from any viewing angle, with 2-D MPR (multi-planar reformat) or 3-D volume rendering. Furthermore, one could use the jigsaw to either specify a region to be removed (exclusive) or a region to be visualized (inclusive). This is very useful, for example, in removing the spine in abdominal studies.

In most cases, the jigsaw is used only to produce a suitable extruded volume but, occasionally, refinement is necessary. Multiple application of the jigsaw is equivalent to computing the intersection between two or more extruded masks. Figure 2 illustrates how applying the jigsaw tool in two different orientations can refine a 3-D shape. The

implementation of the jigsaw is dependant upon the implementation of the renderer and the masking operation.

(ii) The Interpolator

The Interpolator uses a technique known as shape interpolation to generate masks for the volume rendering operation. Traditionally, shape interpolation has been used to create binary volume for solid rendering. However, the present approach is to employ this technique for feature extraction. The inputs to this tool are inexact region(s) of interest obtained from manual contouring of a few 2-D images (typically, 10 for a dataset with 200 images). The advantage of using this tool is that it is much more time-effective than the painstaking process of manually outlining the object(s) of interest on every image of the dataset.

The interpolator works by first converting the binary input slices to distance images. In these images, each pixel value represents the shortest distance between the pixel and the nearest edge. A positive value indicates that a pixel is inside the region of interest and a negative value means outside. Cubic spline interpolation is used to compute the missing data in the gaps, resulting in a series of intermediate slices, which must then undergo a thresholding operation to produce the final mask. A suitable shape interpolator implementation is described in S. P. Raya and J. K. Udupa, "Shape-Based Interpolation of Multidimensional Objects", IEEE Transactions on Medical Imaging, vol. 9, no. 1, pp. 32-42, 1990, incorporated herein by refernce.

Both the tools discussed above are used to define volumes of interest (VOI). VOI are irregular volumetric regions that can be used to define the region where the volume rendering visualization is applied. The VOI does not have to exactly match the structure of interest, but rather loosely contain it or exclude the obstructing anatomies. The direct volume rendering process, by using an appropriate opacity transfer function, will allow for the visualization of all the details contained in the VOI. The main benefit of this type of approach is that the creation of the VOI is very time-effective, and it can be applied to a variety of different modalities. For example, as shown in Figure 3, to extract the brain in an MR dataset the user roughly outlines the brain contour on a few slices and then lets the Interpolator tool generate the VOI. This is particularly significant on MRI datasets because the underlying physics is

no compatible with the use of opacity transfer functions in volume rendering for tissue separation.

In most clinical protocols there is the need to extract multiple features, or to separate in different sections a part of the anatomy. This means that it is preferable to have the ability to apply multiple VOIs to a single dataset. The volume renderer of the present invention solves this problem in a very general way, by providing the user with the option to define multiple masks and also with options for assigning a different opacity and color transfer function to the voxels included in each region. This essentially defines a local classification, where the color and opacity of a voxel does not only depend on its gray level value, but also on its spatial position.

As discussed above, the brain can be easily extracted from an MR dataset using the Interpolator tool. A second VOI can be created using the jigsaw tool in exclude mode starting from the original dataset. The head and the brain are two distinct VOIs and different opacities, color settings, and even rendering modes can be applied to them. In Figure 4 they are rendered together, allowing one to simulate regular as well as unusual extensive craniotomy.

In both CT and CTA dataset most of the tissues have distinct Hounsfield values, and so the opacity can be used to selectively visualize a particular tissue type. Nevertheless the sculpting tools can be used to enhance the visualization, by either applying local classification to specific anatomical part or to maintain references in the dataset. Figure 5 shows the sculpting techniques applied to both a CT and CTA dataset. In the image shown in Figure 5(a), the aorta has been identified with the Interpolator tool by drawing very few contours on the original slices. The rest of the dataset has been partitioned in two regions using the jigsaw tool. In these three regions different classification has been applied and all the VOI have been rendered using shaded volume rendering.

In the Figure 5(b) the dataset has been similarly partitioned using the Jigsaw tool. A different classification has been assigned to the two VOI to show soft tissue in one and bone in the other. Both VOI are rendered in unshaded mode using the original density values.

A preferred embodiment of the present system has been implement with the Imaging Applications Platform (IAP) application development toolkit, an object-based software development package specifically designed for medical imaging applications development. IAP is designed for visual data processing, and therefore all the important 3-D data structures are voxel based. The two most important are: binary solid, and gray level slice stack. Several objects in our visualization pipelines accept both types of data, providing a high level of flexibility and inter-operability. For example, a binary solid can be directly visualized, or used as a mask for the volume rendering of a gray level slice stack. Conversely, a gray level slice stack can be used to texture map the rendering of a solid object or to provide gradient information for gray level gradient shading.

A gray level slice stack (slice stack) is generally composed of a set of parallel cross-sectional images (rasters) acquired by a scanner, and they can be arbitrarily spaced and offset relative to each other. Several pixel types are supported from 8 bit to 16 signed with floating point scaling factor. Planar, arbitrary and curved reformats (with user-defined thickness) are available for slice stacks. Slice stack can also be interpolated to generate isotropic volumes or resampled at a different resolution. They can be volume rendered with a variety of conventional compositing modes such as: Maximum Intensity Projection (MIP) and unshaded or shaded volume rendering. Multiple stacks can be registered and rendered together, each with a different compositing mode.

To generate binary solids, a wide range of segmentation operations are provided: simple thresholding, geometrical regions of interest (ROIs), seeding, morphological operations, etc. Several of these operations are available in both 2-D or 3-D. Binary solids can be reconstructed from slice stack, and shape interpolation can be applied during the reconstruction phase. Once binary solids have been reconstructed, logic operations (e.g. intersection) are available together with disarticulation and cleanup functionality. Texture mapping and arbitrary visibility filters are also available for binary solids.

The following is a list of relevant objects in this context:

Object	Function
Cproj	Performs several different types of volume rendering
Vol	Compute all the view independent pre-processing
Ss	Collects a set of 2-D images (Raster) into a volumetric dataset
Bv	Collects and operates on binary volumes (bitvol)
Recon	Collects a stack of bitmaps and produces a bitvol
Sinterp	Interpolates, resample and re-map a slice-stack
Lim3	Manages a set of axis-aligned 3-D clipping parameters
Extbv	Generates extruded or "tubular" bitvol based on one or more irregular regions of interest
Pvx	Manage a pixel value mapping
Pref	Manages progressive refinement in rendering by collecting the output of multiple rendering pathways
Rf	Manages the arbitrary reformat of a slice-stack
Bp	Performs operations with bitmaps
Geom2	Manages 2-D geometry consisting of a list of lines, polygons, arcs and markers
Tx3	Manages a 3-D coordinate transformation
Ras	Manages the data associated with a 2-D image

A volume-rendering pipeline can be formed by instantiating and connecting few of these objects. The basic pipeline is shown in Figure 6.

Images that are managed by Ras objects are aggregated to form a stack in the Ss object. Performing interpolation on the images forms a volume. The Cproj object then renders this volume, with location and orientation specified by the Tx3 object. Objects that implement the sculpting tools are normally connected to the Vol object as inputs.

Sculpting with the Interpolator requires that parallel regions of interest (ROI) be specified in various locations in volume space. To assist the drawing of these ROIs, many parallel cross sections of the volume have to be exposed to the user. This is achieved by connecting the Ss object to a Rf object that interpolates the input slice stack onto a cut-surface. The cut-surfaces can be at any locations and orientations. Moving a cut-surface along its normal axis

allows the user to seethe various cross sections of the volume. At each cross section, the user specifies a ROI geometry that will be managed by a Geom2 object. The location and orientation information required by the reconstruction process can be obtained from the location and orientation of the viewport.

In the following example, the user draws a number of ROIs that are parallel to the input rasters. The Geom2 objects send the ROIs in bitmap form to a Recon object, which constructs a binary volume (bitvol) using the shape interpolation algorithm described above. When the Vol object receives this bitvol data, it ensures that only parts of the volume lying inside the bitvol are visible.

To construct the bitvol for the jigsaw tool, the Extbv object is used. Similar to a Recon object, this object exports a clipping bitvol to a connected Vol object. This object is actually a meta-object that consists of a combination of Bp and Recon objects.

The ROI specified by the user is processed by the Extbv object to produce a set of bitmaps (managed by Bp objects) that are used to construct the extruded bitvol. The Extbv object requires that the orientation and the location of the ROI in volume space be specified. As in the Interpolator tool scenario, the information can be obtained by mapping its location and orientation from viewport space to volume space.

The object-based architecture of the Processing Server allows the above sculpting techniques to be combined to produce a powerful and efficient sculpting tool. This is achieved by connecting the objects involved in various ways. One way of mixing the sculpting techniques is to connect both a bitvol exporter (Recon, Extbv) object and a Lim3 object to the Vol object. In this case only parts of the volume that are inside both the bitvol and the bounding box are rendered.

A second way to sculpt is to first duplicate a volume, and then applying various combinations of bitvols, created with the jigsaw and the Interpolator tools, to the two volumes. For example, the image in Figures 4(a) and 4(b) was created by the pipeline shown in Figure 7.

In this case, the volume is duplicated by connecting the Ss object to two Vol objects. The two Vol objects are then rendered by the same Cproj object. Each volume is clipped differently. In the sample image, the head (the lower Vol object) is clipped by a Lim3 object, exposing the inner volume (the upper Vol object) that is clipped using shape interpolation.

Bitvols can also be created and modified using Bv objects. This object performs various logical operations such as adding, subtracting and intersecting on input bitvols. The intersection operation is especially useful for iterative refined clipping regions.

The image in Figure 4 was produced by clipping the outer volume with the complement of the extruded bitvol specified by a ROI. The inner volume was clipped using a mask created by the interpolator tool.

In the sculpting techniques of the present invention, VOI and classification can be defined interactively in an iterative faction. In this context, the processing speed is an important factor, because the user preferably needs to visually assess the result of an operation before applying the next. Using a progressive refinement approach, the renderer is able to let the user interactively analyze a clinical size dataset and, if the results are not satisfactory, the operation can be easily undone and repeated. To allow interactive sculpting, each rendering pathway is accompanied by a duplicate that operates on a down-sampled version of the image data.